SPRITE CHASER

#7

Official Newsletter of the #1 ADAM USERS' GROUP

The Sprite Chaser is published by the #1 Adam Users' Group for the enjoyment and furthering the knowledge of its members, in the use of the Adam Computer. Address all correspondence to the Editor, #1 Adam Users' Group, P.O. Box 3761, Cherry Hill, NJ 08034.

All articles are welcome and will be included to broaden our readerships base of knowledge and experience levels. All submissions will be noted and credit given to the author. Articles from this newsletter may be reprinted for use by other users' groups, after permission is granted by us, so long as the users' group in question is an existing viable entity for the benefit of those wishing computer literacy. Users' groups that exist without a membership at large will not be considered a users' group per-se. All articles or letters sent to the Editor for publication are subject to the unrestricted right to edit and comment. DISCLAIMER: The opinions expressed in this publication are the perceptions and opinions of the individual authors and do not necessarily reflect the positions of The #1 ADAM USERS' GROUP. Its publisher, editors, authors or contributors are not liable for any damages, lost profits, lost savings or other incidental or consequential damage arising out of the use of information, or the inability to use any information provided herein. Any inaccurate information will be corrected in a future issue.

The Sprite Chaser is available by subscription for $22.50 per year. It is free to members of the #1 Adam Users' Group. Send all subscription payments and/or change of address to #1 Adam Users' Group, Subscription Service, P.O. Box 3761, Cherry Hill, NJ 08034.

It will be the policy of the #1 Adam Users' Group, not to pass, copy, or sell copyrighted text, diskettes, or any other copyrighted medium thru the group without the permission of the copyright owner either by written permission or through the Copyright Clearance Center, 21 Congress St., Salem, MA 01970.

Membership in the #1 Adam Users' Group does not impart to the general membership any corporate or status to act for the corporation. Membership in the #1 Adam Users' Group is for one year, from the month membership is acquired. Present membership rates are: $15.00/yr. as of September 1, 1986. Rates are subject to change without notice.

The Sprite Chaser is soliciting advertisers at the following rates: FULL PAGE $200 1/2 PAGE $125
1. All submitted advertisements must be printer ready and prepaid. 2. All submitted advertisements are subject to acceptance by publisher. 3. Rates are subject to change without notice.

THE GROUP OFFICERS

PRESIDENT	Jay H. Forman
VICE-PRESIDENT	Stephen George
TREASURER	George M. Rhodes
SECRETARY	Larry Iocco

THE SPRITE CHASER STAFF

EDITOR	Steve George
PUBLISHER	Jay H. Forman
TECHNICAL EDITOR	Dan Decker
RESEARCH EDITOR	Jim Lovenstein....

EDITOR'S MESSAGE

Finally the newsletter has arrived! We are still having problems getting members to contribute articles.... Many thanks again to Greg Daro, who contributed some more information on SmartBasic and using the SmartKeys in Basic. Also to Joe Waters, who has written a dynamite article on using Sprites in SmartBasic 2.0. I have included examples of his programming on PD#37 (SmartBASIC 2.0) to make it easier for you to get started using the newer BASIC.

To enable us to get this newsletter out quicker we are now offering a special: If you write an article or review, send it to us on a DDP or disk, and it is accepted, we will send in return one of the PD disks in exchange (you still will have to send $2 to cover postage, handling, etc.). Our PD library has grown a bit during the past few months. There is a different version of the Cabbage Patch Adventure (different graphics), a superb Music Demo program (PD #38) that features the "theme" music from hit cartridges including some never released (and never seen!!), and 2 more unreleased carts called "Fall Guy" and "Yoke's On You". Both were to be released by 20th Century-Fox, but before they could they dumped there software division. I think Yoke's is unfinished as there seems to be no "finish" to the game. Fall Guy uses the driving wheel or joystick and looks like the "Dukes of Hazzard" cart. Object is riding around, stopping to do stunts (has two different screens) and being chased by cops, etc. I do have a version of "Temple Of Asphai" by Epyx that was going to the full fledged DDP version of the game.. It is *MUCH* different than the cartridge game they released called "Gateway to Asphai"... I am currently researching what to do with it... It can only run from a right directory DDP and is not bug-free but it IS playable! We also have a Demo cart for real collectors of ADAM software. I remember first seeing the ADAM in an Electronics Boutique here in New Jersey and it was play-

ing a looping demo of potential ADAM software and peripherals on it. The cart was chained to the computer!! All of the software and hardware in this demo WAS released! (Not like some of the ads from this period). Pinball Construction Set has been a big hit and some members have a real talent in creating some entertaining pinball games. PD#45 is a stand-alone disk (or tape) of 10 pinball games.. You do not need PCS to use it, although you cannot change the games without PD#31.

For those with modems, People's Link now has an ADAM section (Section Six). The SysOp is David Carmichael and he promises much action in the new section. It does feature a message board and a library. For more information, call "Plink" toll-free at 1-800-524-0100, and in Illinois call 1-312-5200. Well that's it for this issue.. I'll be looking for all those great articles you will be sending me..... Steve George

SPECIAL GRAPHIC EFFECTS IN BASIC 2.0 - by Joe Waters

SPRITES

A sprite is a user defined image, with it's own pattern, color, and graphics plane. BASIC 2.0 has made sprites available to the non-machine language user, but they are not as simple as they seem. To use a sprite, one must do the following. First, the user must define a sprite. This can be done with the Hacker's Guide to ADAM by Peter and Ben Hinkle, or by using a SmartLOGO sprite editor. Either way works fine. This article will not deal highly with that, but with the actual manipulating of sprites. After the binary sprite file has been created, it need only be loaded into memory in a way such as this:

```
0 LOMEM:29000 :REM reserve memory
1 PRINT CHR$(4);"BLOAD SPRITES, A28000" :REM load sprites
```

Next, the DRAW command flag for sprites at 16788 must be turned on, like this:

```
20 POKE 16788,1
```

By POKEing the number 1 into address 16788, you have turned the DRAW command for Shape Tables into a DRAW command for sprites! To turn it back to shape table mode, you must POKE a 1 into address 16788. Should you wish to do so, you could use BOTH sprites AND shape tables simultaneously, by switching between modes. This could come in handy, and can be accomplished like this:

```
30 HIMEM:51455 :REM This makes room in memory for a shape table.
40 PRINT CHR$(4);"BLOAD SHAPE TABLES" :REM This loads the shape tables into their memory location.
```

Now, you have both the sprites and the shape tables in memory. All that remains now is to move them as you please. Here we find a problem with the 2.0 system. A sprite cannot be used unless all of the sprites before it have been drawn on the screen. That is to say that if you want to draw sprite #2, you must first draw sprite #1, and if you wish to draw #4, you must first draw sprites 1,2, and 3, in that order. This could get confusing, unless you make a subroutine to do it for you, like this:

```
50 DIM x(32),y(32),co(32) :REM this creates a block of memory for the sprite x and y coordinates, as well as the sprite status. This routine should only be used once every time you run the program. and... 1000 for p=1 to 32:hcolor=co(p):draw p at x(p),y(p): RETURN:REM this is the sub-routine that is run every time you want to draw a sprite. It is important to note, however, that any one sprite can not be drawn on more than one point on the screen. To draw the sprite, you use a line like this: Let's say the sprite that you want to draw is #10, and you want to draw it at coordinates 125,85 in white, the color code for which is 3. Here's what you do:
60 x(10)=125 :REM this is the horizontal coordinate.
61 y(10)=85 :REM this is the vertical value.
62 co(10)=3 :REM this is the color.
63 GOSUB 1000 :REM this draws the sprite.
```

APPLE COMPATIBILITY

I have noticed one other thing about BASIC 2.0: the DRAW and HCOLOR commands are completely compatible with Applesoft BASIC. In Applesoft, an XDRAW command to a shape table at a location would erase the shape if it is already there, or draw it there if it wasn't. This is different from SmartBASIC 1.0, because it always erased the shape, whether it was there before or not. One other thing about Applesoft and 2.0: the HCOLOR command will now stay at it's present value, even if you go into text mode and come back. This has always been true with Applesoft, but not BASIC 1.0. 1.0 would default to HCOLOR=0 if you went into text mode, but 2.0 will hold it's value no matter what. I hope this has helped some users understand the special effects of this "new" language.

Joe Waters 13009 Gray Hills Rd. NE Albuquerque, NM 87111

****

*** UPLOAD.HLP File for Coleco ADAM Users ***

****

This file is divided into *TWO* sections:

-Section A: ADAMLink II

-Section B: XMODEM

It is intended to get people STARTED, and is NOT a COMPREHENSIVE .DOC File of the complete capabilities of their Software. You should consult your Software .DOC for further information.

****

Section A: ADAMLink II

These are the steps I use when uploading to Compuserve via ADAMLink II:

1.) BEFORE logging on to CIS, insert a blank/formatted disk/datapack into either the 'default' drive or one of the other drives.

2.) AFTER logging on to CIS and 'navigating' to the proper Data Library (in your FORUM, type "DL #" at the "FUNCTION" prompt), type "UPL *ame.ext*".

3.) From the CIS Protocol menu, choose the "CAPTURE" option.

4.) When asked by CIS if the file is an "ASCII File", answer "YES".

5.) When asked by CIS if you want to be "prompted for each line", type "NO".

6.) Type <WILDCARD>

7.) Type <SK VI> "FILE"

8.) Type <SK I> "TRANSMIT"
9.) Type the <SK >corresponding to the proper drive.
10.) Type the "<filename>"
11.) Type <WILDCARD>
12.) Type <STORE/GET> HINT: If you KNOW in advance that you'll be uploading a file, you can complete EVERY STEP EXCEPT #12, BEFORE you log on to CIS. This will save your online time.

****

Section B: XMODEM

XMODEM refers to the Protocol used by 'most' programs for file transfers. It is superior to the 'Capture' protocol used by ADAMLink II because it utilizes an ERROR DETECTION system, thereby preventing bad uploads/downloads. Here are some steps that will work with 'MANY' programs (such as MEX or MADAM7).

1) BEFORE you log on to CIS, either CHANGE DRIVES by typing a new drive label from WITHIN the program (ie. "B:<CR>"), OR log in a NEW blank/formatted D/DP (usually by typing either "LOG" or "RESET").

2) AFTER you've logged on and 'navigated' to the proper FORUM, go to the Data Library (DL) you're going to Upload to (type "DL <#>" at the Forum "FUNCTION" prompt).

3) Type "UPL <filename.ext>".

4) Choose the "XMODEM/MODEM7" option from the Protocol menu.

5) IF the file you're going to upload is an ASCII file (ie. .DOC/.TXT files, SmartBASIC programs, SmartLOGO programs, etc.) then choose the "ASCII" option from the CIS File Type menu, and NOT the "BINARY" option (This will allow ADAMLink II users to download the file).

6) When CIS sends the message "Ready for XMODEM transfer", exit to 'Command Mode'. This is done by typing "<control> E" in MADAM7 and "ESC E" in MEX. (NOTE: The "ESC" key used by MEX varies. To find out your "ESC" key, type "STAT ESC" in Command Mode... Usually it will be "<control> J").

7) In Command Mode, type "S <filename.ext><CR>". Then, you'll 'usually' receive a message as to the "Send Time" (in minutes), # of records, etc.. This is also good information to include in the "DESCRIPTION" of the file you give CIS, AFTER uploading. That way, those who download the file in the future will know in advance approx. how long it will take.

****

SCRUNCH

Here is an interesting program that frees up storage in SmartBASIC files by eliminating excess "whitespace" in programs...

```
10   REM SCRUNCH...Deletes excess spaces in REM and DATA statements.
20   REM  Written by W. R. Faris, 6/09/85
30   d$=CHR$(4):REM <control-D>
40   cr$=CHR$(13):REM <return>
50   quote$=CHR$(34):REM  "
60?  d$; "mon c"
70   INPUT "ENTER FILE NAME: "; file$
80?  d$; "rename "; file$; ", SCRX$ "
90?  d$; "open SCRX$ "
100? d$; "open "; file$
110? d$; "read SCRX$ "
120? d$; "write "; file$
130  ONERR GOTO 460 :REM Error will occur at end of file.
140  b$="": c$=""
150GET a$
160  b$=b$+a$
170IF a$<>cr$ THEN 150
180  i=0
190  i=i+1
200  a$=MID$(b$, i, 1)
210   IF a$>="0" AND a$<="9" THEN c$=c$+a$: GOTO 190
220  b$=MID$(b$, i)
230  i=1
240   IF LEFT$(b$, 4)="REM " THEN c$=c$+"REM ": i=i+4:GOTO 420
250   IF LEFT$(b$, 5)="DATA " THEN c$=c$+"DATA ": i=i+5:GOTO 330
260  a$=MID$(b$, i, 1)
270  c$=c$+a$
280  IF a$=quote$ THEN GOSUB 360
290  IF a$=":" THEN  b$=MID$(b$, i+1):GOTO 230
300  IF a$<>cr$ THEN  i=i+1:GOTO 260
310? c$;
320  GOTO 140
330  a$=MID$(b$, i, 1)
340  IF a$=" " THEN  i=i+1:GOTO 330
350  GOTO 260
360  REM  Handle strings in quotes.
370  I=i+1
380  a$=MID$(b$, i, 1)
390  c$=c$+a$
400  IF a$<>quote$ THEN 370
410  RETURN
420  a$=MID$(b$, i, 1)
430  IF a$=" " THEN  i=i+1:GOTO 420
440  c$=c$+MID$(b$, i)
450  GOTO 310
460? d$; "close SCRX$ "
470? d$; "close "; file$
480  e=ERRNUM(0)
490  IF e=5 THEN ? d$; "delete SCRX$ ":END
500?"ABNORMAL END OF PROGRAM":?"ERRNUM = ";e
510? "ORIGINAL FILE IS NOW 'SCRX$ ' "
520  END
```

****

POKES=PEEK=CALLS= & ADAM PROGRAM HINTS

= = ALL LISTED HERE ARE FOR SMARTBASIC 1.0 = =

[All numbers will need the command "POKE" added before them to work right!]

= =

1146,(ASCII DEC LETTER CODE to change the right bracket)

12185,(128-239 This will let you make the program lines up to a length of 239 characters long)

15824,216:15830,8:15831,55:15832,19(This will fix your copy of SmartBASIC 1.0 so that it only has one space between the DATA or REM statements)

16821,(drive reset - D1=8, D2=24, D5=4 and D6=5)

16953,(ASCII DEC LETTER CODE to change cursor from underscore)

16958,16:16993,8:16995,16(this will increase the TEXT window in the graphic modes to eight lines from the default mode of four text lines. Changing modes (AKA.

going from the GR mode to the HGR or TEXT mode) will reset the text window to its four line default setting)
17000,1(to stop cursor from flashing. Poke a "0" or use a TEXT command to get cursor flashing again)
17059,(TEXT background color)
17115,(TEXT normal letter color)*16+(TEXT normal screen color)
17126,(TEXT inverse letter color)*16+(TEXT inverse screen color)
17164,(ASCII DEC CODE this will fill the primary TEXT screen with any of the codes you select)
17198,(This will set the display area of the bottom of the screen [default setting 23])
17199,(This will let you change the right margin being displayed on the screen [default setting 30])
17201,(This will let you change the top of the screens display area [default setting 00])
17202,(This will let you change the left margin display area on the screen [default setting 01])
17291,(cursor blinking speed with "0"= steady on to "255"=slowest rate of flashing)
17302,255(This will disable the "CONTROL-P" to reenable it poke in 16)
17954,(0-63 will change the pitch of the bell sound of the CHR$(7) command. The lower the number the higher the pitch!)
17963,(0-255 will change the duration of the bell sound of the CHR$(7) command. The lower the number the shorter the duration of the sound.)
18607,(GR background color)
18633,(GR window color)*16+(GR window color)
18711,(GR normal letter color)*16+(GR normal letter color)
20183,3:20187,14:20188,2:20189,0:20190,0(Will let you change the command CATALOG to a shorter command CAT)
25431,(HGR background color)
25471,(HGR window color)*16+(HGR window color)
25568,(HGR normal letter color)*16+(HGR normal letter color)

= =

ALL LISTED HERE ARE FOR SMARTBASIC 2.0

[all numbers will need the command "POKE" added before them to work right!]

= =

16771,(ASCII DEC CODE by poking this code here it can then be displayed by being CALLed by 16770)
17184,(TEXT border color)
17240,(Normal text color)*16+(Normal text color)
17251,(Inverse text color)*16+(Inverse background color)
19361,3:19365,14:19366,02:19367,00:19368,00(This will let you us the shortend CAT command in place of the CATALOG command)

= =

ALL LISTED HERE ARE FOR SMARTBASIC 1.0

[all numbers will need the command "CALL" added before them to work right!]

= =

17046 (Will cause all of your screen colors to be printed in secondary screen memory)
27407 (Read from VRAM)
27420 (Write to VRAM)
64560 (System reset, checks all drives for media to load)
64605 (Hard init, all systems reset)
64638 (Initiate a status request command)
64641 (Initiate a keyboard status request)
64644 (Status of printer)
64647 (Status of data drive)
64650 (Scan active devices)
64659 (Reset keyboard)
64662 (Reset printer)
64665 (Reset data drive)
64743 (this will send you to the Typewriter mode of of SmartWRITER)

= =

ALL LISTED HERE ARE FOR SMARTBASIC 2.0

[all numbers will need the command "CALL" added before them to work right!]

= =

16770 (Will call what ever ASCII DEC CODE that was POKE'd into 16771 and display it on the screen)

= =

APPLE TO ADAM CONVERSIONS = = *ALL COMMANDS LISTED HERE ARE FOR SMARTBASIC 1.0*

= =

APPLE COMMAND = = = =	ADAM COMMAND = =
HAND CONTROLLER	PDL(x)
CALL -198 (Bell sound)	? CHR$(7);
CALL -868 (Clears line)	HTAB 1:?:VTAB VPOS(0)
CALL -3288	RESUME
PEEK(36) (Cursor position)	POS(1) or PEEK(17002)
PEEK(37)(Vert. cursor position)	VPOS(1) or PEEK(17001)
PEEK(216)	NOT USED BY ADAM
PEEK(219)#256+PEEK(218)	PEEK(16127)#256: & PEEK (16126)
PEEK(222)	ERRNUM(1)
PEEK(-16336),PEEK(-16352)	Cassette & speaker ignore
PEEK(-16384) (Keyboard)	PEEK(64885)
POKE 36,x (Move cursor)	HTAB x
POKE 216,0	CLRERR
POKE 232,LS:POKE 233,HS (Shape table address)	POKE 16766,LS: POKE 16767,HS
POKE -16363;character # (Load keyboard buffer)	POKE 16150,255 : & POKE 64885,character
APPLE SCREEN SIZE =40	ADAM SCREEN SIZE =31
APPLE HGR SCREEN SIZE=291	ADAM HGR SCREEN SIZE=255 to convert x coordinates to ADAM multiply by 255/291
PROG. LINE LENGTH=255	ADAM PROGRAM LINE LENGTH=128

****

HELP!

A number of people have mentioned that they are having difficulties in trying to download files from this Data Library. They are not sure what protocol they should be using, etc. Here are the steps I use for downloading.

1) Using the "BROwse" command, find the file you want to download.

2) When you are asked if you want to Read, Download, or go to the Top, enter "r",but DO NOT PRESS RETURN.

3) Next, press <WildCard> to return to the command mode.

4) Next press <IV FILES>.

5) Next press <II RECEIVE>.

6) Next, press the Smartkey for the drive that you want

to save that file to.
7) Next, enter the name that you want to save that file under on your disk/ddp.
8) Next press <VI DONE>.
9) Next press <WildCard> to return to the terminal mode. When you do so, your screen will be blank, that's OK.
10) Next press <RETURN>. The file will start to be displayed on your screen, as this happens it will also be placed into a buffer in your ADAM, every so often, this buffer will be dumped to your disk/ddp.
11) When you have received the entire file, you will see "Key <ENTER> to continue". At this point press <UNDO> to close the file on your disk/ddp.

You have just downloaded a file. Now you can go on to another file and do the same, if you wish. Remember not to give as a filename one that already exists on the disk/ddp that you are using. Remember, also that 14 pages is the most you can hold in one file. If it's a long file that you are downloading, watch the page count at the bottom of the screen.
I hope this file is a help for those having troubles using ADAMLink II.

****

HERE'S ANOTHER NOTE ON DOWNLOADING:

For those of us who find BROwsing through the directory tedious, there is another way to download files using ADAMLink II. Simply, thus: At the Top level of the Data Library, and if you know the filename.ext of the file you want, type "DOW filename.ext/proto:capture" This will send the file to you in a continuous stream, which can be interrupted with a Control-S. This can be useful when changing files, disks/ddp, etc. The DL will give you a couple of messages before actually sending the file. When you see the message, "Opening capture buffer...", type <Control>-S because, after this message, the DL starts sending data. Press <WildCard> then to leave terminal mode and set up your capture buffer. After finishing that, press <WildCard> again to re-enter terminal mode and then type Control-Q. This starts the sending of data to you, and you will see it displayed on the screen as it is sent. The same procedure can be used to upload files to the DL with no trouble. To learn how to set up your capture buffer, either read the ADAMLink II manual, or read the "DOWNLO.HLP" file in this very Data Library. I hope this information proves useful to those others out there in ADAM-Land.

Happy ADAMing!

****

Downloading with XMODEM Protocol (using MEX, MADAM7, etc.)

****

This is an Addendum that I've added in answer to many recent requests on how to download using either MEX, MADAM7, or some other program using XMODEM protocol. It is intended as a BRIEF guideline.. Consult your program .DOC file for more information.
1) BEFORE you log on to CIS or any other online service with the intention of Downloading files, be sure to either CHANGE DRIVES (This is 'usually' done merely by typing in the drive label (ie. "B: <CR>"), OR insert a BLANK/ disk/data pack into your 'default' drive (drive A:) and 'logging in' that blank media (usually by typing EITHER "LOG" or "RESET", depending on the program).
2) AFTER you've logged on to CIS and used found the file you wish to download (using either (BRO) or (DIR)), type "DOW <filename.ext">.
3) When the CIS Protocol menu is displayed, choose the "XMODEM/MODEM7" Option.
4) After CIS displays the "Initiating XMODEM transfer" message, you must exit to "Command Mode". This is done in MADAM7 by typing "<control> E" and in MEX by typing "ESC E". (NOTE: The "ESC" key used by MEX may vary and can be found by typing "STAT ESC" in Command Mode... 'Usually' it will be "<control> J").
5) In Command Mode, type "R <filename.ext>". This will initiate the XMODEM transfer. Then, just sit back and watch... and wait.

****

SMARTKEY MENU

The SmartKey label routine can be used in a menu system that branches to subroutines. Start by typing in the label program. Use lines 1 to 99 for defining variables any lomem or himem statements. Line 100 will be the central point of your program; all branching will be from this area. Starting at line 200, build your menu using "PRINT", "HTAB", "VTAB" etc. The routine provided allows the SmartKeys to be used unshifted or shifted, so you have up to 12 menu items possible. Here is a sample routine that you can modify to meet your needs:

```
200 VTAB 2:  HTAB 3 : PRINT "I     'MENU TEXT GOES HERE'
210 VTAB 3:  HTAB 3 : PRINT "II    'MENU TEXT GOES HERE'
220 VTAB 4:  HTAB 3 : PRINT "III   'MENU TEXT GOES HERE'
230 VTAB 5:  HTAB 3 : PRINT "IV    'MENU TEXT GOES HERE'
240 VTAB 6:  HTAB 3 : PRINT "V     'MENU TEXT GOES HERE'
250 VTAB 7:  HTAB 3 : PRINT "VI    'MENU TEXT GOES HERE'
260 VTAB 8:  HTAB 3 : PRINT "^I    'MENU TEXT GOES HERE'
270 VTAB 9:  HTAB 3 : PRINT "^II   'MENU TEXT GOES HERE'
280 VTAB 10: HTAB 3 : PRINT "^III  'MENU TEXT GOES HERE'
290 VTAB 11: HTAB 3 : PRINT "^IV   'MENU TEXT GOES HERE'
300 VTAB 12: HTAB 3 : PRINT "^V    'MENU TEXT GOES HERE'
310 VTAB 13: HTAB 3 : PRINT "^VI   'MENU TEXT GOES HERE'
320 GET Q$ ;*Get input from the operator*
```

At this point some explanation is needed. When the user presses any key on the keyboard, Q$ will be set to the ASCII character assigned to that key. This is converted to the decimal equivalent using the BASIC command ASC($). The unshifted SmartKeys return the value 128 to 133 and when shifted return the value 137 to 141. This is used with the BASIC command 'ON x GOSUB ' to branch.

```
330 X = ASC(Q$) : IF X  128 OR X  142 THEN HOME : GOTO 200
```

**Check the input to be sure one of the SmartKeys has been pressed; redo the menu if not.*

There are several ways to branch to a subroutine based on the input. You can repeat 'IF x = n then gosub nnn' 12 times to branch according to the selection. There is a much better way, use the 'on x gosub' statement and its built in error trapping. to continue:

```
340 x = x -127
350 ON X GOSUB L,L,L,L,L,L
```

Replace the 'L' with the line numbers to branch to for keys I through VI, unshifted. If a shifted key has been

pressed, control branches to the next line.

```
360 x = x - 4
```

If a key other than the shifted SmartKey is pressed, pass to the next line.

```
370 ON X GOSUB L,L,L,L,L,L
```

Replace the 'L' with the line numbers to branch to for keys I through VI, shifted.

```
380 GOTO 100
```

The 'ON x GOSUB' routine branches to the line number in the series according to the value of x. If x = 1, then the first line is used; x = 2, then the second line number, and so on. If x is 0 or is greater than the number of lines in the list, program control drops through and the 'ON x' line is ignored. This works to our advantage. In line 330, x can take on a value between 128 and 142. The program subtracts 128 from this value in line 340. The value of x is now between 1 and 15 depending on which key was struck and whether it was shifted or not. When the routine branched to from the menu is done, return control to the menu routine by 'RETURN'. Program control will resume with the next line after the gosub statement. If this is line 360, the x will be reduced by 4 and will be less than 1 so control drops to line 380 which displays the menu.

SMART KEY LABELS

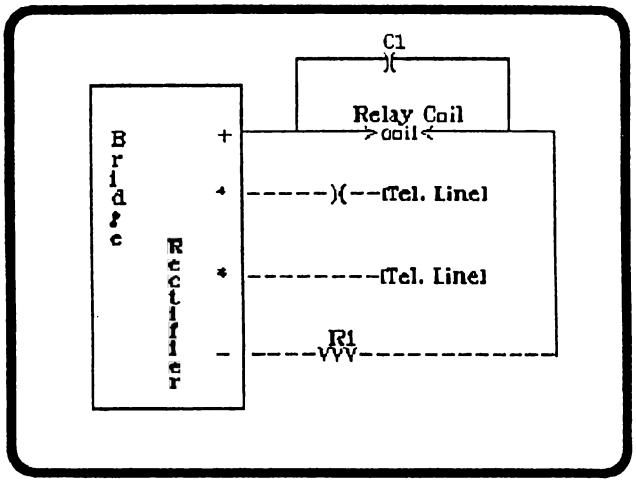
When I bought the Hacker's Guide II, one of the first things I used where the key macros. One of the problems I had, was trying to remember what each of the key functions were. So, I wrote a short program that labels the keys and remains on the screen until I 'TEXT' the screen. To clear the screen I 'HOME'. Scrolling starts at the top of the labels, so no matter how much I fill the screen, my labels are always there. Here is the program with comments following each line:

```
100 POKE 17126, 27: POKE 17059, 14 :POKE 17115,21 * Change the border, text, screen and inverse colors. You may change these to your preferred colors
110 TEXT * Clear the screen
120 INVERSE : POKE 16956, 0: POKE 16957, 32 * Set the labels for inverse colors; reset the left and right margins to the maximum allowable
130 HTAB 0: VTAB 23 * Move to the top display line for the SMARTkeys
140 PRINT " |nnnn|nnnn|nnnn|nnnn|nnnn|nnnn|" * This is the label line. Substitute your text for the 'nnnn', keep all spaces as shown
150 HTAB 0: VTAB 24 * Move to the start of the second label line
160 PRINT " |nnnn|nnnn|nnnn|nnnn|nnnn|nnnn|"; * This is the second label line. BE SURE TO KEEP THE LEADING SPACE AND THE ENDING SEMI-COLON
170 POKE 16956, 1:POKE 16957, 31 * Reset the left and right margins to the system default values
180 POKE 16959, 21:POKE 16993, 22 * Set new bottom of screen and scroll position
190 NORMAL : HOME * Set display to normal and clear the screen
```

It is important to keep the display lines exactly as shown. This insures that the color will be entirely across the screen and both lines will display done

****

HARDWARE Project: ADAM AutoAnswer Interface



(Note:Wire colors below are those from a standard Coleco Vision joystick controller.If you have a spare one, it's the best source for the cable for PORT 1)

```
                    [relay]        [D1]
[to pin 5] <--------> SPST <------(|  )--------> [to pin 6]
(green wire)        [contacts]                   (blue wire)
```

(Note:To complete the diagram use a ruler and felt tip pen to make solid lines on the dash lines and I's. PORT 1 is numbered as below.)

```
            PORT 1
(looking at rt. side of console)
        \  1 2 3 4 5  /
         \ . . . . . /
          \ . . . . /
           \6 7 8 9/
```

An Explanation

Only when the phone line rings, an AC voltage is passed through C2, rectified by the bridge and appears as DC at the + and - leads. This DC voltage is then reduced to about 8-12 volts by R1 then applied to the coil which causes the relay contacts to close. This simulates the right trigger being pushed. For convenience Radio Shack #'s are listed below:

R1- 5.6K ohm 1/2watt resistor.....271-031
C1- .22 uf Capacitor.....272-1070
C2- 4.7 uf Capacitor.....272-1012
Relay- 12 Volt Reed Relay.....275-233
Bridge Rectifier.....276-1171 or 276-1152
D1- 1N914/1N4148 Diode.....276-1122

The values above are the ones that worked best for me and should work well on all standard phone lines. Because of the relay, ADAM isn't directly connected to the phone line and is "safe". Looking at the diagram the connections marked [Tel.Line] are the ones to be connected to the red and green wires from your phone line. With

modular phone jacks these are the two center contacts. (Polarity doesn't matter.) C2 can be put inline with either of the leads from the bridge rectifier marked ~ (or AC). The + side of the bridge goes to one side of the relay coil (see back of package), and to one side of C1. The other side of the coil and C1 then go to one side of R1. The other side of R1 then goes to the - of the bridge rectifier. On to the Joystick connection... The only critical thing here is the polarity of the diode (D1). Looking at the diode you will see a band toward one end (I tried to show it in the diagram), the opposite end (no band) MUST be connected to pin 6 (blue wire). From there the end with the band goes to one of the relay's switch contacts. The other switch contact goes to pin 5 (green wire). It's a good idea to check out the circuit before you plug into PORT 1. With the phone line connected and ringing, tick the + lead of an ohm meter into pin 6 and the - lead into pin 5 at the end of the J/Stick cable, and check for continuity. You should see the meter going on and off. If not check that D1 is correct! Then check for 8-12 volts across the relay coil going on and off. If necessary try changing the value of R1 by about 1000 ohms at a time.

Memory Usage and BASIC

Computers are very simple digital devices and only function in terms of on or off (1's or 0's). Every activity must be expressed in these terms. All Z80A executable commands are expressed as a series of 1's and 0's. When you boot SmartBASIC you are actually loading a series of machine language subroutines that make up the BASIC interpreter. The BASIC interpreter translates your high level commands into machine executable code. The interpreter can be thought of as a program with many subroutines that convert BASIC commands to 1's and 0's. As BASIC commands are executed various machine language routines are called upon to provide the final output. The interpreter occupies the first 27407 bytes of memory ending at LOMEM. Starting at 53361 (HIMEM) are the I/O buffers and the operating system. That leaves you with 25954 of usable RAM for your program code and variables. The interpreter consists of several sections. Beginning at 256 is the command table. This is a list of BASIC command words and their subroutine locations. The next section consists of many of the subroutines and tables which keep track of input and screen display. The central loop is next. This is like an "INPUT" statement in that the system is waiting for input, flashing the cursor to let the user know that the machine is ready for input. Following the central loop are the math functions, math tables and tape (disk) commands.

When you begin typing in a program, the central loop takes the input from the keyboard, checks to see if it is a control character (^C etc.) and if it is not, it stores the information in a buffer. When the loop encounters ^M (RETURN), it begins to translate the information in the buffer. The buffer is then checked to see if it contains an ASCII word. If so, the BASIC command tables is scanned for a match. If a match is found, the remainder of the buffer is checked for proper syntax. If the syntax is correct, the command is executed. If no match is found in the BASIC command table, the math and tape tables are then checked. If no match is found an error message is displayed. If the syntax is wrong, an error message will display. All this takes place very quickly. In fact for some commands this will seem to be instantaneous. If the first item found in the buffer is a program line number, the interpreter works much the same way, but with some important variations. As a match is found, the ASCII word is replaced with a one byte token. For example, the command 'PRINT' is replaced with the token '07'. Each command word and some math and tape commands have a unique token value. This can save a considerable amount of memory, since only the token is stored, not the entire ASCII line. After the syntax is checked, some of the syntax characters and spaces are stripped out and the data is stored with the token. The program lines are stored beginning at HIMEM and working downward in RAM toward LOMEM. The program lines are stored without the line numbers, so they are in the same order the were typed in. The line numbers are stored in a separate table right after the program lines. When the user inserts a line, the line number table is reordered, not the actual program lines. The line number table keeps track of the program lines and points to their location in RAM.

When a BASIC program is saved to disk or tape, the interpreter reads the program tables (line numbers and program code). The tokens are translated back into the ASCII equivalents and then written to the DDP or disk. The opposite occurs when that same program is loaded from DDP or disk. The translation is what takes the time in loading and saving. Programs like Turboload bypass this translation and save the program in its tokenized form. This can save DDP and disk space and reduce the save and load times dramatically since translation is not done.

The LOMEM and HIMEM statements can be used to reserve space for user data or machine language routines. The space reserved is below LOMEM or above HIMEM. Keep in mind that the BASIC interpreter ends at 27407, so do not write below that value. The I/O buffers start at 53361, so do not write above that value. When a HIMEM statement is encountered, the program tables are moved downward accordingly. They will start at the new HIMEM location. The same is true for the variable table and LOMEM. Variables are stored beginning at LOMEM and working upward toward HIMEM. Memory is allocated when a variable is first used or when a DIM statement is executed. As variable space is needed, it moves upward toward HIMEM. When the bottom of the program tables are encountered by the variable table, you are out of memory! Variables consume a tremendous amount of memory. There are some things that you can do to conserve memory, but more on that in the next issue.

****

Let's Program in SmartLOGO!

Here is something we have yet to touch upon! LOGO is a Greek word that means "word". The computer lan-

guage was originally written to teach computer novices the basics of programming. It provides IMMEDIATE feedback and since being ported to microcomputers, can be used to view graphics and sound with a minimum of commands. Coleco's SmartLOGO was actually done by LCSI in Canada, one of the companies involved with the original LOGO language. In fact Seymour Papert one of the founders wrote most of the tutorial of the manual! It is an excellent version (I have used AppleLOGO, Terrapin LOGO for Apple, Commodore LOGO) and can give you a good idea what the ADAM hardware is capable. You can easily program the sound chip, 32 sprites and use all colors! Here is a program called "INSTANT" that will move the turtle around without having to hit return after every command.....

Logo Instant *by C. Kolander*

Instants are single keystroke superprocedures that allow preschoolers to use logo easily. Instead of having to learn logo primitives and then type them all out, instant lets you enter a single key and draws them "instantly". This instant is a very slightly changed version of an instant you can find in the book APPLE LOGO by Harold Abelson,Byte Books. The only thing you may want to do is save the program before running it, as 'record' makes a list (history) so that if you make a mistake you can type 'u'(undo) which will clear your last move, this would be saved with the program and would save the drawing you last made. If you press '^' before making your next drawing it will clear the history list so you can start over. If you would like to increase or decrease the repertoire of this instant just change the Command procedure to fit your needs. Have fun.

```
TO PLAY
COMMAND
PLAY
END

TO PRINFO :ANS
IF :ANS = "Y [PRINTER]
END

TO INFO
PR [INSTANT]
PR []
PR [DO YOU WANT A PRINT OF]
PR [COMMANDS?]
PRINFO RC
CS
PR [KEY COMMAND]
PR [- - - - - - - - - -]
PR []
PR [F FORWARD 10]
PR [R RT 30]
PR [L LT 30]
PR [B BACK 10]
PR [P PENUP]
PR [D PENDOWN]
PR [A FILL]
PR [1 - 0 SETS PENCOLORS]
PR [Z SMALL CIRCLE]
PR [X MED CIRCLE]
PR [V LARGE CIRCLE]
PR [H HT]
PR [S ST]
PR [U UNDO]
PR [C CS]
PR [^ CLEAR DRAWING]
NOPRINTER
END

TO INSTANT
CS
INFO
PR []
PR [TO START - - TYPE DRAW]
END

TO RUN.ALL :COMMAND
IF :COMMAND = [] [STOP]
RUN FIRST :COMMAND
RUN.ALL ( BUTFIRST :COMMAND )
END

TO UNDO
IF :HISTORY = [] [STOP]
MAKE "HISTORY BUTLAST :HISTORY
CS
RUN.ALL :HISTORY
END

TO RECORD :ACTION
RUN :ACTION
MAKE "HISTORY ( LPUT :ACTION :HISTORY )
END

TO COMMAND
MAKE "COM READCHAR
IF :COM = "F [RECORD [FD 10]]
IF :COM = "B [RECORD [BK 10]]
IF :COM = "R [RECORD [RT 30]]
IF :COM = "L [RECORD [LT 30]]
IF :COM = "C [RECORD [CS]]
IF :COM = "H [RECORD [HT]]
IF :COM = "S [RECORD [ST]]
IF :COM = "1 [RECORD [SETPC 1]]
IF :COM = "2 [RECORD [SETPC 2]]
IF :COM = "3 [RECORD [SETPC 3]]
IF :COM = "4 [RECORD [SETPC 11]]
IF :COM = "5 [RECORD [SETPC 5]]
IF :COM = "6 [RECORD [SETPC 6]]
IF :COM = "7 [RECORD [SETPC 7]]
IF :COM = "8 [RECORD [SETPC 8]]
IF :COM = "9 [RECORD [SETPC 9]]
IF :COM = "0 [RECORD [SETPC 10]]
IF :COM = "P [RECORD [PU]]
IF :COM = "D [RECORD [PD]]
IF :COM = "A [RECORD [FILL]]
IF :COM = "Z [RECORD [CCIRCLE 10]]
IF :COM = "X [RECORD [CCIRCLE 20]]
IF :COM = "V [RECORD [CCIRCLE 30]]
IF :COM = "U [UNDO]
IF :COM = "^ [ERN "HISTORY]
END
```

```
TO DRAW
CS ST
MAKE "HISTORY []
PLAY
END

TO RCP :R
RT 5
FD :R * 3.14159 / 18
RT 5
END

TO CCIRCLE :R
MAKE "PEN? PEN
HT
PU FD :R
RT 90
PD REPEAT 36 [RCP :R]
LT 90 PU
BACK :R
SETPEN :PEN?
ST
END

MAKE "STARTUP [INSTANT]
MAKE "COM "^
MAKE "PEN? [PENUP 1]
```

THE HOWITZER GAME in SmartBASIC

```
1 LOMEM :29000
5 NOTRACE
10 TEXT:HOME: pi=3.141593/180
20 DIM yy(1000), sq(1000), b(1)
30 VTAB 8: n$="H O W I T Z E R":GOSUB 9999
35 VTAB 12: n$="Downloaded from Compuserve":
GOSUB 9999
40 VTAB 13: n$="Information Service":GOSUB 9999
45 VTAB 15: n$="Revised by J Raymond":GOSUB 9999
50 VTAB 16: n$="1985-1986":GOSUB 9999
55 VTAB 20: n$="Sound? (y/n)":GOSUB 9999
56 GET sound$:IF sound$"Y" AND sound$"y" AND
sound$"N" AND sound$"n" THEN 56
60 FOR x=1 TO 100: sq(x)=(x/10)*(x/10):NEXT x
70 HGR:HOME
71 VTAB 22: n$="<press any key to begin>":GOSUB
9999: j=PEEK(64885)
73 x=RND(2): r=PEEK(64885)
74 x=COS(221): x=RND(2)
75 IF r=j THEN 73
76 HOME
80 VTAB 21:? "ANGLE: 0"; TAB(11); "WIND: 0"; TAB(20);
"ANGLE: 0"
100? SPC(1); "BAGS: 0"; SPC(12); "BAGS: 0"
110 HCOLOR =1: np=0
120 p=RND(1)*20+139: hg=RND(1)*70+25
130 FOR i=0 TO 1000: yy(i)=p:NEXT i
140 FOR i=159 TO p STEP -1:HPLOT 0, i TO 255,i:NEXT
i
150 FOR i=107 TO 147: hx=SIN(((i-106)*4.5+180)*pi)
*hg
160 yy(i)=hx+p
170 HPLOT i, hx+p TO i, 159:NEXT i
180 wind=INT(RND(1)*10):IF RND(1).5 THEN wind=-
wind
190 VTAB 21:HTAB 17:? ABS(wind); :HTAB 0
200 HCOLOR =3:IF wind=0 THEN 240
210 HPLOT 107, 10 TO 147, 10
220 IF wind0 THEN HPLOT 107, 10 TO 120, 5:HPLOT
107, 10 TO 120, 15:GOTO 240
230 HPLOT 147, 10 TO 134, 5:HPLOT 147, 10 TO 134, 15
240 b(0)=RND(1)*15+55-5*ABS(wind)
250 b(1)=RND(1)*15+185+5*ABS(wind)
260 FOR i=0 TO 1:FOR j=b(i)-3 TO b(i)+3:HPLOT j, p-
5 TO j, p:NEXT j
280 HCOLOR =2:NEXT i
290 REM
310 VTAB 24:HTAB 0:? SPC(30); :VTAB 24:HTAB
20*np:? "ANGLE?"; :GOSUB 2000: an=z
330 IF an OR an175 THEN GOSUB 1000:GOTO 310
340 VTAB 22:INVERSE
350 VTAB 21:HTAB 7+19*np:? SPC(4);
360 VTAB 21:HTAB 7+19*np:? an; :HTAB 0
370 VTAB 24:HTAB 0:? SPC(30); :VTAB 24:HTAB
20*np:? "BAGS?"; :GOSUB 2000: bg=z
380 IF bg OR bg40 OR bgINT(bg) THEN GOSUB
1000:GOTO 370
390 VTAB 22:INVERSE
400 VTAB 22:HTAB 7+19*np:? SPC(6);
410 VTAB 22:HTAB 7+19*np:? bg; :HTAB 0
420 VTAB 23:HTAB 0:? SPC(30);
425 ON (sound$="N") OR (sound$="n") GOTO 430:
duration=5: inc=0:GOSUB 10000
430 bg=bg*10:IF np=1 THEN an=an+180
440 py=p-5: px=b(np): iv=0
450 iv=iv+1
460 x=bg*iv*COS(an*pi)/10+b(np)-wind*sq(iv)
470 y=bg*iv*SIN(an*pi)/10: y=p+(np-( NOT np))*y+
16*sq(iv)
490 IF y>p+7 THEN 510
500 IF x>b(np)-5 AND x<(np)+5 THEN 570
510 ym=(ABS(py-y)/((ABS(px-x))+1E-04))*SGN(y-py)
520 st=SGN(x-px): rn=x-px: cc=0
530 IF n=1 THEN cc=rn: rn=0: st=-st
540 l=px+cc: m=py+ym*ABS(cc)
550 IF yy(ABS(l)) THEN 610
560 cc=cc+st:IF ABS(cc-rn)ABS(st) THEN 540
570 ON (x254) OR (x) OR (y) GOTO 580:FOR cv=1 TO
7:HCOLOR =3:HPLOT x, y:NEXT:HCOLOR =0:HPLOT
x, y
580 px=x: py=y
590 IF yy(ABS(x))y THEN l=ABS(x): m=yy(ABS(x)):
GOTO 610
600 GOTO 450
610 IF ABS(x-b(0)) OR ABS(x-b(1)) THEN 660
620 IF x OR x254 OR y THEN 650
630 HCOLOR =0:FOR i=1 TO 25: bx=l-2+RND(1)*4:
by=m+RND(1)*2:HPLOT bx, by
640 ON (sound$="N") OR (sound$="n") GOTO 650:
duration=100: inc=-1:GOSUB 10000
650 np=1-np:GOTO 290 660HCOLOR =3:FOR i=x-10
TO x+10 STEP 2:HPLOT ABS(i), p-(RND(1)*10) TO x,
p:NEXT i
665 ON (sound$="N") OR (sound$="n") GOTO 670:
duration=200: inc=0:GOSUB 10000
670 VTAB 23:? SPC(6); "Another game? (y/n) ";
680 GET a$:IF a$"Y" AND a$"y" AND a$"n" AND a$"N"
```

```
THEN 680
690 IF a$="Y" OR a$="y" THEN 70
700 TEXT:HOME:END
1000 VTAB 23:HTAB 0:? CHR$(7); SPC(10); "BAD VALUE!"; SPC(10); :FOR dl=1 TO 1000:NEXT dl:RETURN
2000 c=1: l$="0"
2010 GET a$: a=ASC(a$)
2020 IF a=13 THEN z=INT(VAL(l$)*10)/10:RETURN
2030 IF a45 AND a: AND a47 AND c THEN l$=l$+a$: c=c+1:? a$; :GOTO 2010
2040 IF a163 THEN 2010
2050 IF c=1 THEN l$="0":GOTO 2010
2060 l$=LEFT$(l$, LEN(l$)-1): c=c-1:? a$; :GOTO 2010
9999 HTAB (15-(LEN(n$)/2)):? n$:RETURN
10000 REM explosion sound routine
10010 GOSUB 10100:REM init sound values
10020 POKE chip%, 228:CALL sound%
10030 FOR loud=240+inc TO 255
10040 POKE chip%, loud:CALL sound%
10050 FOR delay=1 TO duration:NEXT
10060 NEXT loud
10070 RETURN
10100 DATA 58,102,109,211,255,201
10110 sound%=28000
10120 chip%=28006
10130 FOR address=sound% TO sound%+5
10140 READ byte%
10150 POKE address, byte%
10160 NEXT address
10170 RESTORE
10180 RETURN
```

****

#1 ADAM USERS' GROUP -Public Domain Sfotware Exchange

Software Exchange "Public Domain" means these programs can be freely modified and exchanged. Under no circumstances can they be "sold" for profit. Usually, programs MUST retain a "remark" statement crediting the source of the program. Common sources of public domain software are large databases (i.e. "Compuserve", "The Source"), other PD Exchanges (NIAD, ECN, AUG), listings from magazines, newsletters, and from individual users contributing to the exchange. Our Exchange includes elements from all of the above. Because of space I cannot describe all the programs. I will try to highlight the few that one should really have. *If you have any questions about individual programs you may call me at 201 679 6102 (please do not call collect). You may also send me a SASE to:* **STEVE GEORGE 67 STEVENS AVENUE, OLD BRIDGE, NJ 08857**. SmartWriter HELP files have an "H" designation. CP/M help files have a ".DOC" or ".TXT" extension. These can be read by using the "type" command. You can print them out by hitting the Control key + the P key (this toggles the printer on and off in CP/M). Following is an example. A. type filename.ext <RETURN> On BASIC disks any file with an "H" designation can be read in SmartWriter. These are usually 'README" files.

Rules

1) Make all checks payable to "*STEVE GEORGE*". Postal or Money orders are filled immediately. If you send a personal check expect a 8-10 day delay. Stamps are also accepted. Most disks will go out NEXT DAY!! (I hate to wait for software too!)

2) Questions about programs will be only answered if you send me a SASE.

3) For an updated list of programs available send a SASE + $1 (to cover Xerox costs).

4) If you have nothing to contribute, a $9 donation is required. I will ONLY supply disk format. This covers handling, postage, and purchase of disk. You also have the option, if you can supply the disks, of sending me blank, FORMATTED disks + $5 per disk. *If you do not have a disk drive, send a LORAN, M.W. Ruth "PLAIN LABEL", or COLECO DDP + $7 to cover handling and postage. (Victory an FastForward DDPs are unreliable).* I will also include an updated disklist with your order.

5) If you have programs to donate, include a README file + $2 to cover postage and handling. I will copy your files off and then put the requested disk/DDP on YOUR disk. I will accept DDP only if they are the above-mentioned DDPs. You can send me stamps if you want to.

6) You will not always get the same disk you sent. We use single-sided, double - density disks only.

Data Pack or Disk - Disk will be sent unless requested in DP.

Disk #1 - CP/M: MBOOT3 - Simple binary file upl/dl, Filter - from ADAM CP/M manual (Chpt 4), FICOPY - file disk copier modifier written in CP/M, can edit anything on a disk

Disk #2 - CP/M: Cart Utility, ROMHEX - Dump carts contents in hex/ascii format, Cartcopy - copy carts to a disk. Creates.com file. Carts can be run and moved easily., RSTDSK -Tired of typing CTRL-C? Use rstdsk! This disk also has the SOURCE CODE to the cart copier for easy modification.

Disk #3 - BASIC: BALLOON - great sprite game, MOD - use w/Packcopy to edit DDP/Disks, 3D Tic tac toe, Dueling Cannons - hires game, HELLO -access any disk easily, Brickout - Different than Bonanza version, Lunar Lander - classic text game, LANDER -shape table graphics, STAR TREK - joystick controlled text adventure, LIFE - graphic population study, BLACKJACK - text, HANOI -graphic game, Paint - has save feature, Eliminator - Hi-res graphic shoot-em' -up, Many other files....

Disk #4 - CP/M: Master Catalog System - Entire cataloguing system for your disks... Access any file from your CP/M disks easily

Disk #5 - BASIC: BACKUP.TAPE - can be modified to backup disks, CARTCOPY -(Not as good as Multicart, but it does work!), TAPEDITOR -Create versions of any sftwre, YAHTZEE, CPMFILTER - to "clean" files from CP/M to BASIC, CROSSFIRE - graphic game, GET EM' - graphic game SMARTDRAW! - w/save feature, COLORTST - test screen colors of monitor Many other files...

Disk #6 - CP/M: LU.COM - Library Utility, SQ.COM - "squeeze" all files 40% with this utility, USQ.COM - "unsqueeze" files that have been squeezed, DUU.COM - Ward Christiansen's famous CP/M disk utility, LOGALL.COM/.DOC, BUGS.COM, MEMMAP.COM, SORT.COM

Disk #7 - CP/M: Z80 Programmer - Specialized, use w/#8

Disk #8 - CP/M: SCRNCHOP.COM, Z80 Programmers II, Specialized use w/#7

Disk #9 - CP/M: EBASIC.COM - A basic for CP/M..One of the first! Complied NOT interpretive, EBASIC.DOC - Documentation for EBASIC, ERUN.COM - The COMPILER for EBASIC, NSWP.COM/.DOC, EDIT.COM/.DOC - Better than Adam's ED.COM!

Disk #10 -Telecommunications Package: MADAM7 - (Use ADAM Modem... Has auto dial/ans feature.. Better than ADAMLINK II!), MEX (Modem Executive) - Also self-dialing Programmable keys function, Phone book, SCNCRNCH.COM - yet another screen chop Docs for all

Disk #11 - Z80MAC.LBR - (Need LU.COM to extract files) Fully featured assembler, w/docs A must for serious programming, SOURCE.LBR - Use w/Z80MAC to create Z80 code or change 8080 code to Z80. Can create .ASM files from .COM files! - Includes docs

Disk #12 - CP/M:CP/M ADVENTURE-Classic comes to ADAM includes scr. chop

Disk #13 - BASIC: Personal Finance Planner, CHRUNCHER - similar to Turboload, compresses BASIC programs to run 10 times faster!, COLORCALC DISKINIT, CISPICT- read hires pictures done by other computers *Includes some examples, HIRESEASEL, MOVEFILES, UTILITYMOD, HI - Q

Disk #14 - BASIC: EVILTEXT, BATTLESHIP, JOINFOUR, CHECKERS, SERPENT, MATHBLS, MENUPRG, SPACE, SCHEDULE, MAYANQUILT All above include doc files

Disk #15 - BASIC: (from NIAD's Exchange.. This is a fantastic disk... probably best ADAM PD BASIC disk ever!!!), BACKGROUND, BLOCKS, CRAYONS CANON3 -music, KBEXAMPLE, NOISEMAKER, FDUMP -Fantastic!, BICOLOLR - Background/text color changes, INSTANT, FATFONT -makes ADAM look like PC JR in 40 col mode, SPRITEDIT, CONVERT, ICECREAM, FDUMP2 USA MAP.. AND much more!

Disk #16 - BASIC: ROCK w/DOC, HOCKEY w/DOC, GAMESBUS, MASTERMIND, STRONGHOLD w/DOC, MINEFIELD w/DOC, TEXTROCKET, METRIC CONVERTER -converts common metric measurements to USA measures and vice versa

Disk #17 - CP/M: MUMPS - a database file system for ADAM VERY complete you need #18 too.

Disk #18 - CP/M: MUMPS II - use with disk #7 VERY complete!

Disk #19 - CP/M: SPLIT.COM - Break large data file into small ones, I/OMAP.COM, BASEBALL.COM, SECURE.COM, /.COM - link commands, HALLEY-follow the comet, INVENTY.COM, Documentation included

Disk #20 - CP/M: DU-V89 - a more recent version of DUU, TYPTRYT.LBR - a CP/M typing tutor, MLOAD24, XCCP.LBR, XCCPUTIL.LBR, MARKET.COM, XRAM - test your RAM disk memory

Disk #21 - CP/M: SMALL-"C"-a 'C' compiler for the ADAM!, UNIXTOOL.LBR

Disk #22 - DDTF.COM,.DOC,.LBR, FIND.LBR, HOLD.COM,.DOC, KILL.COM, SUPERZAP.LBR, ENCRYPT.LBR, TRIVIA.LBR - Play trivial pursuit

Disk #23 - CP/M: BISHOW.ASM,.COM, LDIRR.LBR, QK21.PRM, YANCTERM.PRM, CERTIFY.LBR, PROBE.LBR, Z80 CHESS, (sue LU.COM or NULU.COM to extract the files)

Disk #24 - CP/M: COBOL for the ADAM! Includes ALL one would need to program COBOL on ADAM 1 Full disk!

Disk #25 - CP/M: ORGCHOP.COM, FINDBAD.COM, CONT.COM, PRINT.COM, MAKBATCH.COM, DISPLAY.COM, SLOAD.COM, CATPAGE.COM, CATLBR.LBR, MCAT - 41.LBR

Disk #26 - CP/M: FORTH for the ADAM!! 1 Full disk with documentation * everything one needs to program FORTH on the ADAM!!

Disk #27 - LOGO: A full disk of logo utilities and games

Disk #28 - CP/M: A full featured BBS program - RBBS (Remote Bulletin Board System) Squeezed 159K!!!

Disk #29 - Coleco-Never released... Jeopardy

Disk #30 - Coleco-Never released... Troll's Tale

Disk #31 - Coleco-Never released... Best of Electronic Arts *Hard Hat Mack * Pinball Construction Set

Disk #32 - Coleco-Never released... Super Sub Roc-90K!! Much better than cart. version!!

Disk #33 - Video Hustler.. Never released

Disk #34 - Test Cart.. Test system peripherals, printer, DDP, etc.. Need 64K expander, CP/M to run...

Disk #35 - BASIC- Misc BASIC files from NIAD/AUG Users Group Library

Disk #36 - BASIC - BBSKit (Updated Bulletin Board Sys. for Adam in BASIC)

Disk #37 - Coleco-Never released... SmartBASIC #2 includes DOC's with almost 30K of Help files.

Disk #38 - Coleco Music Demo - 25 "songs" from Coleco cartridges (some never released)

Disk #39 - NIAD LOGO disk ... to be used with SmartLOGO (almost full w/ demos, etc.)

Disk #40 - Coleco DDP Speed Tester - Test the speed of your tape drives

Disk #41 - Cabbage Patch Adventure 128K - Although I do not think this is a full 128K cart, it is DIFFERENT than the regular version.

Disk #42 - Never released cartridges ... FALL GUY/YOKE'S ON YOU ... Two carts never released (they do contain bugs but they do not affect the playability of the games ... Fall Guy - You can use the driving wheel Yoke's On You - Protect your chicks from snakes, gophers, crows, foxes, etc....

Disk #43 - Misc SmartBASIC #2 .... Includes many utilities, games, hires pictures FULL DISK!

Disk #44 - Coleco Demo Cart - A demo cartridge given out to various dealers showing future software titles for the Adam

Disk #45 Pinball Construction Games - A bootable disk with 10 pre-made, excellent games on it. You do not need PCS (PD #31), but you cannot edit these games without it.

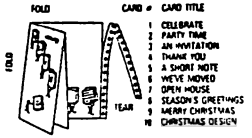
Disk #46 CP/M EBASIC Games disk Many classic games converted to EBASIC (you do not need EBASIC to use this disk) Lunar, Gunner, StarTrek, etc.. (Full disk)

Disk #47 CP/M Pascal for the AdamProgram in PASCAL with this disk!

NEW COLOR MAIL FOR JUST $11.95*

Spruce Up Your Business, Social, Personal Messages with Colorful Pre-printed Greeting Card Announcements. (Fan Folded-Tractor Fed)

Use With Any Computer-Printer-Any Word Processing Program.



*Special Sampler Pre-pack Offer
50 Cards-40 Envelopes Only $11.95

COMPUTER LABELS/ACCESSORIES/SUPPLIES

5 1/4" DISKS	(SS,DD 25/$13.25-10/$ 6.40) or (DD,DD 25/$15.00-10/$ 7.50)
3 1/2" DISKS	(SS,DD 25/$37.50-10/$16.50) or (DD,DD) 25/$42.25-10/$17.95)
Paper T/F-F/F White 9 1/2" x 11", 20lb.	1000/$17.95
Labels T/F-F/F (Address)	1000/$ 5.95
Labels T/F-F/F (Data pack and/or cassette tape)	100/$ 5.95
Labels T/F-F/F for 3 1/2" disks that wrap around disk	100/$8.95
Labels T/F-F/F for 5 1/4" disks that cover full top of disk	100/$4.95
Index Cards - T/F-F/F - 3" x 5"	500/$7.95
Rolodex Cards - T/F-F/F - 2 1/8" x 4"	500/$8.95
Disk Holder - Holds up to 50 disks - anti/static - 3 1/4" or 5 1/2" disks	$11.95
Monitor/TV Stand-360 rotation,up to 12.5 angle	$21.95

ADAM

DATA PACK (Ad a m)	1/$3.95 - 10/$34.50
RIBBON CART.(Adam)	1/$5.50 - 3/$15.00
DAISY WHEEL(Adam) - Italic, Script, Elite, Bookface, Dual Gothic, etc.	1/$5.50
ADAM COVERS - Set with Logo for system	$18.95
ADAM DISK COVER - To match above	$9.95
TRACTOR FEED for Adam printer	$68.95
PRINTER STAND - Front ON/OFF switch	$16.95
POWER PACK - To separate printer/use CPU alone	$29.95
PACKCOPY - Backup SmartBASIC, etc.	$29.95
BLACK GOLD - Look for oil. Survey-profits-fun	$19.95
BOUNTY HUNTER - Text adventure	$14.95
EBU - SmartBasic - Data Pack or Disk - SmartBASIC replacement PLUS	$21.95
HACKER'S GUIDE TO ADAM - Disk or DP - Vol I or II	$17.95
SIGNSHOP - Design & Printing System(DP)	$20.95
MicroWORKS (DP) 5 programs that work along or together (1. Wordprocessor/text editor 2. Database 3. Spreadsheet 4. Picture editor 5. Business graphics	$25.95
ProofREADER, Spell Checker (DP) Shipping appromx. June	call $
TurboLOAD - Speeds up loading programs, included FILE ORGANIZER.	$25.95
PRO-GOLF CHAMP (DP/D) Realistic graphic golf game	$14.95
DEMONS & DRAGONS (DP/D) Test your strategic skills in this easy to use role playing fantasy adventure	$19.95
QUEST FOR QUINTANA ROO (DP) Great all graphic games made for ADAM	$11.95

S&H-$2.50 US
S&H-$4.50 CN
US $'s only

VISA
MASTER

M.W. RUTH CO., Dept. 1AD67
3100 W. Chapel Ave.
Cherry Hill,NJ 08002
(609) 667-2526

FREE CATALOG - We stock what we sell, for FAST DELIVERY

BULK RATE
U.S. POSTAGE
PAID
PERMIT #989
CHERRY HILL, NJ

RETURN POSTAGE GUARANTEED

#1 ADAM Users' Group
P.O. Box 3761
Cherry Hill, NJ 08002